would do little to achieve the vision of the National Council of Teachers of Mathematics (2000) that children should be mathematically literate in a world where mathematics is rapidly growing and is extensively being applied in diverse fields. What other approaches help achieve that vision?

Turtle Geometry

Directing the movement of Logo's 'turtle' can also provide challenging learning experiences. In Logo, children give commands to direct an on-screen turtle to move through 'roads' or mazes or to draw shapes. Primary-grade children have shown greater explicit awareness of the properties of shapes and the meaning of measurements after working with Logo (Clements & Nastasi, 1993). For example, while drawing a face in Turtle Math™ (Clements & Meredith, 1994), Nina decided to draw her 'mouth with a smile' with exactly 200 turtle steps (approximately millimeters). Off-computer she wrote a procedure where the sides of the rectangle were 40 and 20 and the sides of each equilateral triangle were 10. She realized that the total perimeter of these figures was 20 short of 200 and changed just one side of each triangle to 20. Running these procedures on the computer, she remarked that changing the length of one side 'messed up' an equilateral triangle and consequently her 'smile'. She had to decide whether to compromise on the geometric shape or the total perimeter. Her final 'mouth' was a rectangle of 200 steps and her 'smile' was an equilateral triangle of 60 steps.

Logo programming is also a rich environment that elicits reflection on mathematics and one's own problem-solving. Students use certain mathematical notions in Logo programming, such as notions of inverse operation. First grader Ryan wanted to turn the turtle to point into his rectangle. He asked the teacher, 'What's half of 90?' After she responded, he typed RT 45. 'Oh, I went the wrong way.' He said nothing, eyes on the screen. 'Try LEFT 90,' he said at last. This inverse operation produced exactly the desired effect.

Other children may need teacher assistance to link their knowledge of mathematics to their computer work as well as Nina did. Teachers can ask children to reflect on their work, especially 'surprises,' when the computer does something other than what they want it to do. Such reflection can promote greater self-monitoring and may encourage them to find computer 'bugs' themselves (Clements et al, 1993).

Logo sometimes can be difficult for young children to comprehend. However, when the environment is gradually and systematically introduced to the children and when the micro-worlds are age-appropriate, they do not show signs of any problems (Clements, 1983-84; Brinkley & Watson, 1987-88b; Cohen & Geva, 1989; Watson et al, 1992; Howard et al, 1993; Allen et al, 1993). Thus, there is substantial evidence that young children can learn Logo and can transfer their knowledge to other areas, such as map-reading tasks and interpreting right and left rotation of objects.

Why should Logo be especially helpful in developing spatial concepts? From a Piagetian perspective, students construct initial spatial notions not from passive viewing, but from actions, both perceptual [3] and imagined, and from reflections on these actions (Piaget & Inhelder, 1967). These are critical foundations; however, unless they are mathematized [4], they remain only intuitions. Many experiences can help children reflect on and represent these actions; research indicates that Logo's turtle geometry is one potent type of experience. Logo environments are in fact action based. These actions are both perceptual – watching the turtle's movements – and physical – interpreting the turtle's movement as physical motion that could be performed oneself. By first having children form paths and shapes by walking, then using Logo, children can learn to think of the turtle's actions as ones that they can perform; that is, the turtle's actions become 'body syntonic.'

But why not just draw it without a computer? There are at least two reasons. First, drawing a geometric figure on paper, for example, is for most people a highly proceduralized and compiled process. Such a procedure is always run in its entirety. This is especially true for young children, who have not re-represented the sequential instructions that they implicitly follow. Then, they cannot alter the drawing procedure in any substantive manner (Karmiloff-Smith, 1990), much less consciously reflect on it. In creating a Logo procedure to draw the figure, however, students must analyze the visual aspects of the figure and their movements in drawing it, thus requiring them to reflect on how the components are put together. Writing a sequence of Logo commands, or a procedure, to draw a figure 'allows, or obliges, the student to externalize intuitive expectations. When the intuition is translated into a program it becomes more obtrusive and more accessible to reflection' (Papert, 1980, p. 145). That is, students must analyze the spatial aspects of the shape and reflect on how they can build it from components.

And they do. Primary-grade children have shown greater explicit awareness of the properties of shapes and the meaning of measurements after working with the turtle (Clements & Nastasi, 1993). They learn about measurement of length (Sarama, 1995; Campbell, 1987; Clements et al, 1997) and angle (du Boulay, 1986; Kieran, 1986; Olive et al, 1986; Frazier, 1987; Clements & Battista, 1989; Kieran & Hillel, 1990; Browning, 1991). One microgenetic study confirmed that students transform physical and mental action into concepts of turn and angle in combined off- and on-computer experiences (Clements & Burns, 2000). Students synthesized and integrated two schemes, turn as body movement and turn as number, as originally found (Clements et al, 1996). They used a process of psychological curtailment in which students gradually replace full rotations of their bodies with smaller rotations of an arm, hand, or finger, and eventually internalized these actions as mental imagery.

These effects are not limited to small studies. A major evaluation of a Logo-based geometry curriculum included 1624 students and their teachers and a wide assortment of research techniques, pre- and post-paper-and-pencil

testing, interviews, classroom observations, and case studies (Clements et al, 2001). Across grades K–6, Logo students scored significantly higher than control students on a general geometry achievement test, making about double the gains of the control groups. These are especially significant because the test was paper-and-pencil, not allowing access to the computer environments in which the experimental group had learned and because the curriculum is a relatively short intervention, lasting only six weeks. Other assessments confirmed these results, and indicated that Logo was a particularly felicitous environment for learning mathematics, reasoning, and problem-solving.

As an example, consider a class of first graders, constructing rectangles with blocks, string, pencils and papers, pegboards, sticks, and computers (Clements et al, 2001). 'I wonder if I can tilt one,' mused a boy working with Logo. He turned the turtle, drew the first side, then was unsure about how much to turn at this strange new heading. He finally figured that it must be the same turn command as before. He hesitated again. 'How far now? Oh, it must be the same as its partner!' He easily completed his rectangle. The instructions he should give the turtle at this new orientation were initially not obvious. He analyzed the situation and reflected on the properties of a rectangle. Perhaps most important, he posed the problem for himself.

Students in another class had explored the notion that a square was a rectangle – a special type of rectangle. They then had created a parallelogram with Logo. One of the students came up to his teacher the next day and said that he was thinking about parallelograms at home. 'Is a rectangle a special parallelogram?' he asked. 'Why do you say so?' 'Because it's just like the rectangle procedure if it had 90° turns.' This conversation shows that the student had used his Logo experiences to extend his thinking about relationships between polygons (Clements et al, 2001).

These studies indicate that Logo, used thoughtfully, can provide an additional evocative context for young children's explorations of mathematical ideas. Such 'thoughtful use' includes structuring and guiding Logo work to help children form strong, valid mathematical ideas. Children do not appreciate the mathematics in Logo work unless teachers help them see the work mathematically. These teachers raise questions about 'surprises' or conflicts between children's intuitions and computer feedback to promote reflection. They pose challenges and tasks designed to make the mathematical ideas explicit for children. They help children build bridges between the Logo experience and their regular mathematics work (Clements, 1987; Watson & Brinkley, 1990/91). These suggestions are valid for most types of open-ended software and will be discussed in a later section.

Further, recent versions of Logo, such as Turtle Math, have built-in features that were designed based on research. As a small example, 'turn rays' (see Figure 1) help students differentiate between the 'turn angle' (exterior angle) and interior angle and help them conceptualize the measure of turns.

Research indicates that such features facilitate mathematical learning (see Clements et al, 2001, chapter 5).
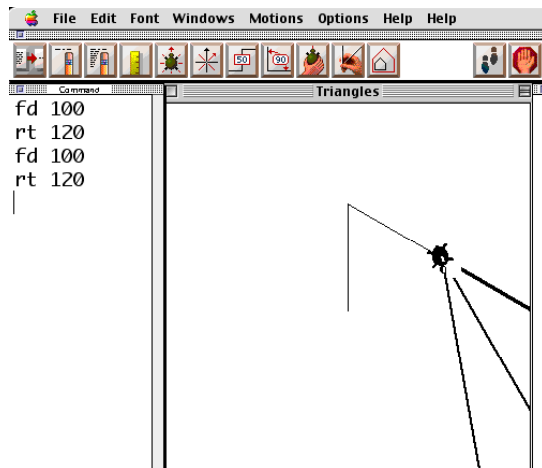

Figure 1. 'Turn Rays'.

In summary, Logo has some unique advantages (Clements & Battista, 1989, 1992) in that it links children's intuitive knowledge about moving and drawing to more explicit mathematical ideas, encourages the manipulation of specific shapes in ways that help students in viewing them as mathematical representatives of a class of shapes, facilitates students' development of autonomy in learning (rather than seeking authority) and positive beliefs about the creation of mathematical ideas, encourages wondering about and posing problems by providing an environment in which to test ideas and receive feedback about these ideas, helps connect visual shapes with abstract numbers, and fosters mathematical thinking (Clements, 1994).

Higher-order Thinking Skills

Computers can also help develop other higher-order thinking skills. Preschoolers who used computers scored higher on measures of metacognition (Fletcher-Flinn & Suddendorf, 1996). They were more able to keep in mind a number of different mental states simultaneously and had more sophisticated theories of mind than those who did not use computers. Several studies have reported that Logo experience significantly increases in both preschool and primary grade children's ability to monitor their comprehension and problem-solving processes; that is, to 'realize when you don't understand' (Clements & Gullo, 1984; Clements, 1986, 1990; Lehrer & Randle, 1986; Miller & Emihovich, 1986). This may reflect the prevalence of 'debugging' in Logo programming. Other abilities that may be positively affected include

understanding the nature of a problem, representing that problem, and even 'learning to learn' (Lehrer & Randle, 1986; Clements, 1990). Along with the increase in metacognitive talk in writing and mathematics activities, there is a substantial argument that computers can foster young children's metacognition.

Problem-solving computer activities motivate children as young as kindergartners to make choices and decisions, alter their strategies, persist, and score higher on tests of critical thinking (Gélinas, 1986; Riding & Powell, 1987). Specially designed computer programs can improve analogical thinking of kindergartners (Klein & Gal, 1992); a variety of problem-solving CAI programs significantly increased first and second graders' ability to generalize and solve mathematics problems (Orabuchi, 1993). Several studies reveal that Logo is a particularly engaging activity to young children, fostering higher-order thinking in children from preschool through the primary grades, including special needs students (Degelman et al, 1986; Lehrer et al, 1986; Clements & Nastasi, 1988; Nastasi et al, 1990). Preschool and primary grade children develop the ability to understand the nature of problems and use representations such as drawings to solve them. When given opportunities to debug, or find and fix errors in Logo programs (Poulin-Dubois et al, 1989), they also increase their ability to monitor their thinking; that is, to realize when they are confused or need to change directions in solving a problem (Clements & Nastasi, 1992).

Unique advantages of computers for fostering higher-order thinking include: allowing children to create, change, save, and retrieve ideas, promoting reflection and engagement; connecting ideas from different areas, such as the mathematical and the artistic; providing situations with clear-cut variable means–end structure, some constraints, and feedback that students can interpret on their own; and so allowing children to interact, think, and play with ideas in significant ways, in some cases even with limited adult supervision (Clements, 1994).

Computer Manipulatives

In one approach, children explore shapes using general-purpose graphics programs or 'computer manipulatives.' Researchers observing such use observe that children learn to understand and apply concepts such as symmetry, patterns and spatial order. For example, Tammy overlaid two overlapping triangles on one square and colored select parts of this figure to create a third triangle which was not provided by the program. Not only did Tammy exhibit an awareness of how she had made this, but she also showed a higher-order awareness of the challenge it would be to others (Wright, 1994). As another example, young children used a graphics program to combine the three primary colors to create three secondary colors (Wright, 1994). Such complex combinatorial abilities are often thought out of reach of young

children. In both these examples, the computer experience led the children to explorations that increased the boundaries of what they could do.

Computer manipulative programs extend general purpose graphics programs in allowing children to perform specific mathematical transformations on objects on the screen. For example, whereas physical base-ten blocks must be 'traded' (e.g. in subtracting, students may need to trade 1 ten for 10 ones), students can break a computer base-ten block into 10 ones. Such actions are more in line with the *mental actions* that we want students to learn. The computer also links the blocks to the symbols. For example, the number represented by the base-ten blocks is dynamically linked to the students' actions on the blocks, so that when the student changes the blocks the number displayed is automatically changed as well. This can help students make sense of their activity and the numbers.

Thus, computer manipulatives can provide unique advantages (Sarama et al, 1996; Clements & Sarama, 1998), including: saving and retrieving work, so children can work on projects over a long period (Ishigaki et al, 1996); offering a flexible and manageable manipulative, one that, for example, might 'snap' into position; providing an extensible manipulative, which you can resize or cut; linking the concrete and the symbolic with feedback, such as showing base-ten blocks dynamically linked to numerals; recording and replaying students' actions; and bringing mathematics to explicit awareness, for example, by asking children to consciously choose what mathematical operations (turn, flip, scale) to apply to them.

Integrated Approach

Of course, several approaches may be combined in one program. Julie Sarama and I designed our *Building Blocks* software to enable all young children to build solid content knowledge and develop higher-order, or critical, thinking. To achieve this, we needed to consider the audience, determine the basic approach to learning and teaching, and draw from theory and research in each phase of the design and development process. Based on theory and research on early childhood learning and teaching (Bowman et al, 2001; Clements, 2001), we determined that the basic approach of *Building Blocks* would be *finding the mathematics in, and developing mathematics from, children's activity*. The materials are designed to help children extend and mathematize their everyday activities, from building blocks to art to songs and stories to puzzles. Activities are designed based on children's experiences and interests, with an emphasis on supporting the development of *mathematical* activity. So, the materials do not rely on technology alone, but integrate three types of media: computers, manipulatives (and everyday objects), and print. Here I will briefly describe some of the computer activities.

Each activity has several levels, often containing quite different tasks. For example, in Double Trouble (see Figure 2), children decorate cookies, counting and adding to produce a given number of chips on each. At Level 1,

children choose a twin cookie with the same number of chips as a given cookie. At Level 2, children make a twin cookie with the same number of chips as a given cookie. That is, they have to make a twin cookie with the same number of chips as a cookie Mrs Double shows them. At Level 3, children decorate cookies with a number given by Mrs Double, so they have to 'count out' the correct number twice, with no 'model.' At Level 4, children play a game in which they tell how many chips have been hidden under a napkin, when, for example, first two and then one more are shown to be placed under the napkin. At Level 5, children count on to 'fix' a cookie that has too few chips; for example, making a '4 cookie' into a '7 cookie.' In Free Explore, children make cookie problems for one another.

Figure 2. 'Double Trouble'.

As another example, in Party Time, children use one-to-one correspondence and counting to help set a table for a party. At Level 1, children get ready for a party by putting one of each item (such as plates, spoons, etc.) on each place mat. At Level 2, the character gets the items out, but asks the child to tell how many are needed. The child must count the place settings at the table. At Level 3, the character switches roles, telling the child how many place settings there are, and asking the child to get out that number of each item needed. In Free Explore, children create their own parties (see Figure 3).

Notice that free explore tasks – basically manipulatives in context – are critical to our design. As a final example, Shape Puzzles invites children to solve outline puzzles by putting together shapes. They use the tools to move the shapes into place. They move through research-based levels, from puzzles that are simple and 'obvious' (see Figure 4) to puzzles that are 'open' and challenging and require that children mentally combine shapes (Figure 5).

Again, there is a Free Explore activity here; children make their own shape puzzles for other children.

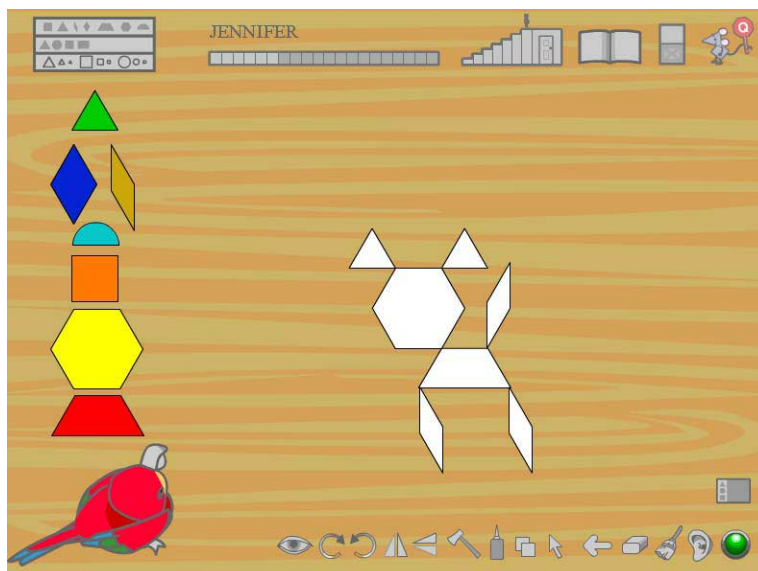Figure 3. 'Free Explore'. Create your own parties.
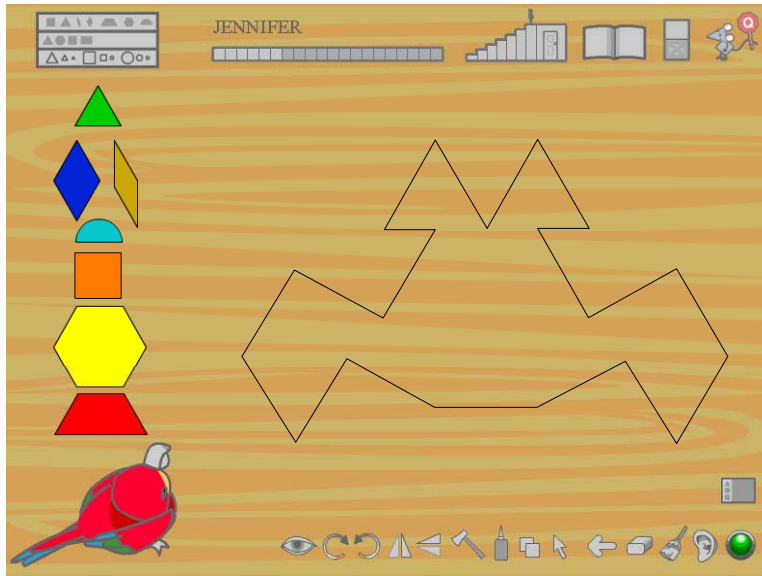

Figure 4. Shape puzzles – simple.

Figure 5. Shape puzzles – challenging.

Thus, children receive numerous opportunities for practice, within meaningful problem-solving contexts that require use of higher-order thinking skills. They also use higher-order and creative thinking in the free explore activities. Initial field testing of our first Building Blocks software product that embodies this integrated approach has been positive. In one study, preschoolers made substantial gains in both the areas of number and geometry (Sarama, in press).

The successful teachers in most of the studies in this entire section were consistently mediating children's interaction with the computer (Samaras, 1991). The importance of the teacher's role is the subject to which we now turn.

Teaching with Computers

Even in preschool, children can work cooperatively, with minimal instruction and supervision, if they have adult support initially (Rosengren et al, 1985; Shade et al, 1986). However, adults play a significant role in successful computer use. Children are more attentive, more interested, and less frustrated when an adult is present (Binder & Ledger, 1985). Thus, teachers may wish to make the computer one of many choices, placed where they can supervise and assist children.

Effective Strategies

Across the educational goals, we find that teachers whose children benefit significantly from using computers are always active. They closely guide children's learning of basic tasks, then encourage experimentation with open-ended problems. They are constantly encouraging, questioning, prompting, and demonstrating. Such scaffolding leads children to reflect on their own thinking behaviors and brings higher-order thinking processes to the fore. Such metacognitively oriented instruction includes strategies of identifying goals, active monitoring, modeling, questioning, reflecting, peer tutoring, discussion, and reasoning (Elliott & Hall, 1997). Whole group discussions that help children communicate about their solution strategies and reflect on what they have learned are also essential components of good teaching with computers (Galen & Buter, 2000).

Two studies show clearly that such scaffolding is critical. In the first (Yelland, 1994), children were only given instructions for specific tasks and then mostly left alone. These children rarely planned, were often off task, rarely cooperated, and displayed frustration and lack of confidence, and did not finish tasks. In the second study, using similar software and tasks (Yelland, 1998), the teacher scaffolded instruction by providing open-ended but structured tasks, holding group brainstorming sessions about problem-solving strategies, encouraging children to work collaboratively, asking them to think and discuss their plans before working at the computer, questioning them about their plans and strategies, and providing models of strategies as necessary. These children planned, worked on task collaboratively, were able to explain their strategies, were rarely frustrated, and completed tasks efficiently. They showed a high level of mathematical reasoning about geometric figures and motions, as well as number and measurement.

Such teaching is difficult. A balance of teacher guidance and children's self-directed exploration is necessary for children to learn to appropriate this new technology (Escobedo & Bhargava, 1991). In designing curriculum around open-ended software, research has shown that children work best when designated open-ended projects rather than asked merely to 'free explore' (Lemerise, 1993). They spend more time and actively search for diverse ways to solve the task. The group allowed to free explore grew disinterested quite soon. Models and sharing projects may also be helpful (Hall & Hooper, 1993). Effective teachers also integrate computers into the ongoing program. They balance and combine on-computer and off-computer activities and discuss computer activities in group sessions.

Although the more structured nature of typical CAI tasks allows research and recommendations regarding the time per day students should use the computer, teachers will have to use their knowledge of their classes and individual students in gauging time on more open-ended activities. Often, 10-20 minutes are not adequate; however, overuse should be monitored. There is some evidence that strict time limits for such activities can generate hostility and isolation instead of the usual positive effects of the computers on